

# Ask the Experts

XR の SIA ライセンス  
使用状況のモニタリング  
(XR FCM - SIA license utilization monitoring)

[2024年3月14日]



# Disclaimer

This document is Cisco Confidential information provided for your internal business use in connection with the Cisco Services purchased by you or your authorized reseller on your behalf. This document contains guidance based on Cisco's recommended practices.

You remain responsible for determining whether to employ this guidance, whether it fits your network design, business needs, and whether the guidance complies with laws, including any regulatory, security, or privacy requirements applicable to your business.

## 免責

この文書は、お客様またはお客様の代理人である認定リセラーが購入したシスコサービスに関連して、お客様が社内業務において使用することを目的としてシスコが提供するシスコの機密情報です。この文書にはシスコが推奨するプラクティスに基づく手引きが記載されています。

お客様は、この手引きを使用するか否かやお客様のネットワーク設計および業務上のニーズにこの手引きが適合しているか否か、さらにはこの手引きが法律（お客様の業務に適用される規制上の要件、セキュリティ上の要件およびプライバシーに関する要件を含みます）に準拠しているか否かを判断する責任を引き続き負います。



## 本日の学習内容：

- FCM ライセンスのコンプライアンスチェック方法
- 保有ライセンスや管理製品のレポート方法
- FCM 関連の各種操作の自動化方法

# 本日の トピック

- ① FCM (Flexible Consumption Model) 概要
- ② SIA/RTM コンプライアンスチェック
- ③ ライセンス・製品インスタンスレポート
- ④ FCM 自動化 1 Smart Licensing APIs
- ⑤ FCM 自動化 2 NETCONF
- ⑥ FCM 自動化 3 EZ-Register
- ⑦ デモ

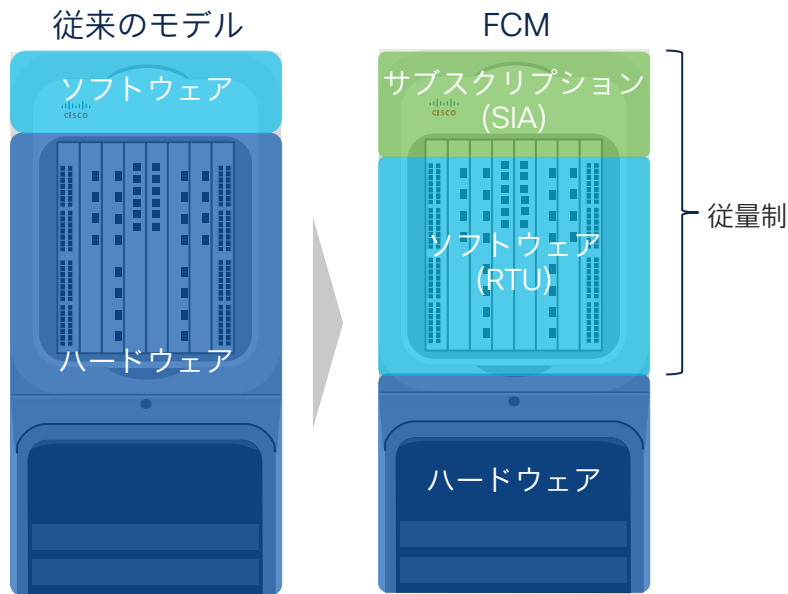
# 本日の トピック

- ① FCM (Flexible Consumption Model) 概要
- ② SIA/RTM コンプライアンスチェック
- ③ ライセンス・製品インスタンスレポート
- ④ FCM 自動化 1 Smart Licensing APIs
- ⑤ FCM 自動化 2 NETCONF
- ⑥ FCM 自動化 3 EZ-Register
- ⑦ デモ

# Cisco IOS-XRの新たなライセンスモデル

## Flexible Consumption Model (FCM)

Flexible Consumption Model (FCM) は、ハードウェアのキャパシティ(帯域幅)やソフトウェアの機能を現在のビジネスニーズやネットワーク要件に合わせて導入、さらに将来のニーズや要件に合わせて拡張および追加できるため、投資を最適化できる購入モデルです



### 初期投資の削減

- 従量制モデルにより、必要なキャパシティ分のRTUを購入
- 初期投資額を削減

使用するキャパシティ分のRTUを購入\*

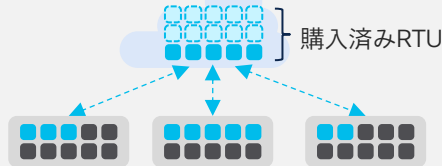


\*各FCM対応ハードウェアの初期購入時には最小限のRTUの購入が必要

### 資産の有効活用

- 使用しているポートの帯域分だけRTUを消費
- お客様のSmart AccountにプールされたRTUを装置をまたがって利用可能
- RTUを無駄なく利用

Smart Account



### 投資保護

- 購入済みのRTUを新しいハードウェアに移行可能
- ライセンスの買い直しが不要

RTUの再利用



旧ハードウェア

新ハードウェア

# FCMライセンス構造

FCMは対応したハードウェアとRTUとSIAを組み合わせて購入します



\* ハードウェア初期購入時に最低3年間の SIA 契約が必要

\*\* ハードウェア初期購入時は各ハードウェアに設定されている必要最小限の Essentials RTU が必要

\*\*\* 一部の製品にはアラカルトライセンスがあります (例 BNG ライセンス、DCO ライセンス)

# 本日の トピック

- ① FCM (Flexible Consumption Model) 概要
- ② SIA/RTM コンプライアンスチェック
- ③ ライセンス・製品インスタンスレポート
- ④ FCM 自動化 1 Smart Licensing APIs
- ⑤ FCM 自動化 2 NETCONF
- ⑥ FCM 自動化 3 EZ-Register
- ⑦ デモ



# ライセンスの管理

全てのライセンスの管理は、Cisco Software Central にて行うことが可能です。  
様々な管理画面に1つのポータルサイトからアクセス可能です。複数のスマートアカウントを管理している場合、スマートアカウントを切り替えてそれぞれの管理画面にアクセスします。



**Cisco Software Central**  
<https://software.cisco.com/>



シスコアカウント  
でログイン

スマートアカウント  
の切替

スマートアカウント abc.com



PAKの管理  
LRP  
License Registration Portal



スマートライセンスの管理  
CSSM  
Cisco Smart Software Manager



SPNA / EAの管理  
EA Workspace



アカウント管理  
Manage Account

SIA/RTM コンプライアンスチェック

# ライセンスの管理 (実際の画面)

Cisco Software Central (<https://software.cisco.com/>)

## シスコスマートソフトウェアマネージャ (CSSM)

ライセンス	課金情報	Available to Use	使用中	代用	残高
④ \$201 Base HW Tracking PID	プリペイド	30	0	-	+30
④ Core & Aggr Advance SIA per 100G for Cisco 8000 series	プリペイド	200	0	-	+200
④ Core & Aggr Advance SW Right-to-Use per 100G for Cisco 8000 series	プリペイド	200	0	-	+200

## スマートアカウントの管理

Virtual Account	Description	Tags
<input type="checkbox"/> EN-Demo		-
<input type="checkbox"/> DCN-Demo		-
<input type="checkbox"/> DCC-Demo		-
<input type="checkbox"/> WAN-Demo		-
<input type="checkbox"/> SP-Demo		-

## ダウンロードと管理

### スマートソフトウェアマネージャ

### ダウンロードとアップグレード

ライセンスを追跡し管理します。従来のライセンスをスマートライセンスに変換します。

新しいソフトウェアのダウンロードや既存ソフトウェアの更新をします。

[ライセンスの管理 >](#)

[ダウンロードにアクセス >](#)

### 従来のライセンス

PAKベースのライセンスやその他のデバイスのライセンス(デモライセンスを含む)を生成して管理します。

[LRP にアクセス >](#)

### スマートアカウントの管理

### EAワークスペース

アカウント情報を更新し、ユーザを管理します。

Cisco Enterprise Agreementを通じて購入したライセンスを生成および管理します。

[アカウントの管理 >](#)

[EA ワークスペースへのアクセス >](#)

### エンタイトルメントの管理

新しいポータルで、eDelivery、バージョンアップグレード、さらに多くの管理機能を利用できるようになりました。

[MCEへのアクセス >](#)

# CSSM - アラート

スマートアカウント全体のアラートが1つの場所に表示されます。メジャーアラートは**赤色**、マイナーアラートは**黄色**、情報アラートは**明るい青色**で表示されます。各アラートに対するアクションを、[アクション] ボタンから実行することができます。

Cisco Software Central > スマートソフトウェアライセンス CS Demo Labs ▼

スマートソフトウェアライセンス SL Product Details サポート ヘルプ

**アラート** | [インベントリ](#) | [スマートライセンスへの変換](#) | [レポート](#) | [設定](#) | [オンプレミス アカウント](#) | [アクティビティ](#)

アラート **39** メジャー | **119** マイナー

利用可能なアクション ▼ | 

<input type="checkbox"/>	重大度	Alert Type	メッセージ	送信元	アクション期限	アクション
<input type="checkbox"/>	✖	製品インスタンスが更新に失敗...	バーチャル アカウント "UDI_PID:Tailf-NSO; UDI_SN:a6a2bb2a-b72a-4cff-b9d3-2be2ab81ee86; " ...	SP-Demo	今すぐ	<a href="#">アクション ▼</a>
<input type="checkbox"/>	✖	製品インスタンスが更新に失敗...	バーチャル アカウント "UDI_PID:Tailf-NSO; UDI_SN:e5a05ae3-a9d7-4a5b-b4aa-67f4611adab3; " ...	SP-Demo	今すぐ	<a href="#">アクション ▼</a>
<input type="checkbox"/>	✖	ライセンスの期限が失効してい...	Smart Software Managerサテライト "dCloud-mekawaba-20230718" は、27 days日間同期していま...	dCloud-meka...	今すぐ	-
<input type="checkbox"/>	⚠	[予約済みライセンスの期限が...	バーチャル アカウント 「"dCloud-pimoh-20221205"」 の製品インスタンス 「"UDI_PID:8201; UDI_S...	dCloud-pimo...	今すぐ	<a href="#">アクション ▼</a>
<input type="checkbox"/>	⚠	[予約済みライセンスの期限が...	バーチャル アカウント 「"dCloud-pimoh-20221205"」 の製品インスタンス 「"UDI_PID:8201; UDI_S...	dCloud-pimo...	今すぐ	<a href="#">アクション ▼</a>

# CSSM - 保有・使用中ライセンス確認

[インベントリ] の [ライセンス] タブには、選択したバーチャルアカウントが保有するライセンスのインベントリ、ライセンスの使用状況（過不足）、およびアラートが表示されます。 これにより、コンプライアンス維持に必要なライセンス数確認およびアクション(ライセンス転送)を実行できます。

Cisco Software Central > スマートソフトウェアライセンス CS Demo Labs

## スマートソフトウェアライセンス SL Product Details サポート ヘルプ

アラート **インベントリ** | スマートライセンスへの変換 | レポート | 設定 | オンプレミス アカウント | アクティビティ

バーチャル アカウント **FCM-Demo** 1 メジャー | 32 マイナー | アラートを非表示にする

全般 **ライセンス** | 製品インスタンス | イベント ログ

利用可能なアクション | ライセンス タグを管理する | ライセンスの予約... |  ライセンストランザクションの表示 |  名前で指定 | タグ別

ライセンス	課金情報	Available to Use	使用中	代用	残高	アラート	アクション
<input checked="" type="checkbox"/> Core & Aggr Advantage SIA per 100G	プリペイド	100	2	-	+98		アクション
<input checked="" type="checkbox"/> Core & Aggr Essentials SIA per 100G	プリペイド	100	2	-	+98		アクション

送信元: Manual Entry  
サブスクリプション ID: N/A

SKU: ESS-100G-SIA-3  
ファミリー: Network Convergence System  
5500 Series

Quantity: 4  
開始日: 2023-Jun-06  
期限: 2033-Jun-03

詳細検索

# CSSM - ライセンスインベントリアラート詳細

ライセンス不足、期限間近、期限切れなどの際にアラートが表示されます。アラートをクリックすると、アラートの詳細及びトラブルシューティング方法を確認できます。

<input type="checkbox"/> ライセンス	課金情報	Available to Use	使用中	代用	残高	アラート	アクション
② Core & Aggr Essentials SIA per 100G for Cisco 8000 series	プリペイド	2	216	-	-214	✖ ライセンスが不足して	アクション ▾
② Core & Aggr Essentials SW Right-to-Use per 100G for Cisco 8000 series	プリペイド	2	216	-	-214	✖ ライセンスが不足して	アクション ▾

**Insufficient Licenses**

✖ バーチャル アカウント "dCloud-mekawaba-20230815" には "Core & Aggr Essentials SW Right-to-Use per 100G for Cisco 8000 series" ライセンスが不足しています。コンプライアンスを再度確保するには、214 個のライセンスが必要です。

**トラブルシューティング** | 詳細

製品インスタンスによって "Core & Aggr Essentials SW Right-to-Use per 100G for Cisco 8000 series" 必要とされる使用量をカバーするには "dCloud-mekawaba-20230815" 製品インスタンスの要件を満たすためです。考えられる解決方法が以下に一覧されています。追加情報は [詳細] タブで確認できます。

**バーチャル アカウント間で転送：**

- 可能な場合は、別のバーチャルアカウントからこのバーチャルアカウントにライセンスを転送してください。 [ライセンスの転送...](#)
- 可能な場合は、1つ以上の製品インスタンスをこのバーチャル アカウントから十分なライセンスのあるバーチャル アカウントに移動してください。

**機能の用途の変更：**

- このライセンス タイプを必要とする製品インスタンスの機能を無効にします。

**購入：**

- 不足をカバーするための追加ライセンスを購入します。
- 可能な場合は、低レベルのライセンスタイプから使用量をカバーするために十分なライセンスタイプへのアップグレードを購入してください。

**Insufficient Licenses**

✖ バーチャル アカウント "dCloud-mekawaba-20230815" には "Core & Aggr Essentials SW Right-to-Use per 100G for Cisco 8000 series" ライセンスが不足しています。コンプライアンスを再度確保するには、214 個のライセンスが必要です。

トラブルシューティング | **詳細**

**ライセンスの詳細**

バーチャル アカウント: dCloud-mekawaba-20230815

名前: Core & Aggr Essentials SW Right-to-Use per 100G for Cisco 8000 series

有効期限: -

現在の数量: 2

使用中: 216

不足: -214

**関連イベント**

以下は、アラートの直前にログに記録された関連イベント (最大 10 件) です。

メッセージ	日時	ユーザ

# CSSM - トラッキングライセンス

トラッキングライセンスは、HW を購入すると指定されたバーチャルアカウントに自動追加されるライセンスで、購入された HW の CSSM への登録有無や実際の稼働数を把握するために使用されます。PID の最後に「-TRK」が付きます。ライセンス費はかかりません。

Cisco Software Central > スマートソフトウェアライセンス CS Demo Labs ▾

## スマートソフトウェアライセンス SL 製品の詳細 サポート ヘルプ

アラート | **インベントリ** | スマートライセンスへの変換 | レポート | 設定 | オンプレミス アカウント | アクティビティ

バーチャル アカウント: **FCM-Demo** 34 マイナー | アラートを非表示にする

全般 | **ライセンス** | 製品インスタンス | イベント ログ

利用可能なアクション ▾ | **ライセンス タグを管理する** | **ライセンスの予約...** |  |  **ライセンストランザクションの表示** |

名前で指定 | タグ別

詳細検索 ▾

ライセンス	課金情報	Available to Use	使用中	代用	残高	アラート	アクション
<input checked="" type="checkbox"/> <b>8201 Base HW Tracking PID</b>	プリペイド	7	5 (4 予約済み)	-	+2		<b>アクション ▾</b>
<input type="checkbox"/> 送信元: サブスクリプション ID: N/A	Manual Entry		<input type="text" value="8201-TRK"/> ファミリ: SpitFire Product Bookings	Quantity: 1	開始日 2023-Jun-06 期限: 2033-Jun-03		

# CSSM - アラートのメール通知

[設定] タブより、発生アラート等に関する日次レポートをメール送信することが可能です。

Cisco Software Central > スマートソフトウェアライセンス

## スマートソフトウェアライセンス

アラート | インベントリ | スマートライセンスへの変換 | レポート | **設定** | オンプレ

### ライセンス設定

製品登録トークン:	有効	変更履歴を表示
特定のライセンスの予約:	有効	変更履歴を表示
ユニバーサルライセンスの予約:	無効化	変更履歴を表示
[インベントリ]タブでのライセンスストラ ンザクションの詳細の表示:	有効	変更履歴を表示
予約済みライセンスの事前インス トル:	無効化	

### 日単位のイベントのサマリー

以下で選択されたイベントを含む日単位の電子メールのサマリーを受信する

電子メールアドレス: mekawaba@cisco.com

バーチャルアカウントの選択: 5 選択済み

追加する電子メールアドレス: カンマ区切り

アラート イベント:

- ライセンス不足 - アカウントでの使用数が使用可能なライセンス数を超過しています
- ライセンスの期限がまもなく失効する - 期限付きライセンスがまもなく期限切れになることを警告します。  
● 90日 ○ 60日 ○ 30日 ○ 14日 ○ 7日 ○ 1日
- ライセンスの期限が失効している - 期限付きライセンスが期限切れになりました。ライセンス有効期限に関する警告が無視されなかった場合のみ表示されます。
- 製品インスタンスが接続に失敗しました - 製品が更新期間内に正常に接続されていません。
- 製品インスタンスが更新に失敗しました - 製品が最大許容更新期間内に正常に接続されていませんでした。
- サテライト同期が期限切れです - サテライトが予約された期間内に同期していません。
- サテライトが登録解除および削除されました - サテライトが90日目の動機に失敗し、削除されています。
- ライセンスが未変換です - 1つ以上の従来のライセンスは、製品インスタンス登録中に自動的にスマートに変換されませんでした。
- ライセンスの消費 - アカウントで使用可能なライセンスの使用 選択

情報イベント:

- 新規ライセンス - 新しいライセンスが正常にアカウントに登録されています。
- 新しい製品インスタンス - 新しい製品インスタンスが正常にアカウントに登録された
- 製品インスタンスが削除されました - 既存の製品インスタンスがアカウントから正常に削除されました
- ライセンスは予約されました - ライセンスが予約されました - アカウントのライセンスが製品インスタンスによって予約されました。

### ステータス通知

サテライト同期ファイルの Smart Software Manager による処理が完了した時点で電子メールを受信する

### Ask Licensing

非アクティブなときにサポートが必要かどうかを確認する

保存 リセット

# 本日の トピック

- ① FCM (Flexible Consumption Model) 概要
- ② SIA/RTM コンプライアンスチェック
- ③ ライセンス・製品インスタンスレポート
- ④ FCM 自動化 1 Smart Licensing APIs
- ⑤ FCM 自動化 2 NETCONF
- ⑥ FCM 自動化 3 EZ-Register
- ⑦ デモ



# ライセンスレポート

[レポート] タブにて、Smart Account / Virtual Account の保有するライセンスに関するレポートを出力できます。

Cisco Software Central > スマートソフトウェアライセンス

## スマートソフトウェアライセンス

アラート | インベントリ | スマートライセンスへの変換 | **レポート** | 設定 | オンプレミス アカウント | アクティビティ

レポート

① レポートを選択

レポート | 使用状況データファイル | ポリシーのレポートिंग | デバイスコントローラの同期ファイル

名前 | 説明

**ライセンス** | ② ライセンスを選択 | View License count and usage over selected Virtual Account

製品インスタンス | 選択したバーチャル アカウントの現在の製品インスタンス

レポート設定が使用されました

Smart Account Name / Domain : CS Demo Labs / cs.cisco.com

Report Title : ライセンス レポート

Report Description: View License count and usage over selected Virtual Account(s)

Generated On: 2024-Mar-03 14:23:23

Report Generated by : mekawaba@cisco.com

⑤ レポート出力例

合計ライセンス使用数				
バーチャル アカウント	ライセンス	購入	Available To Use	使用中
FCM-Demo	8201 Base HW Tracking PID	7	7	5
FCM-Demo	Core & Aggr Advance SIA per 100G for Cisco 8000 s	153	153	0
FCM-Demo	Core & Aggr Advance SW Right-to-Use per 100G fo	147	147	4
FCM-Demo	Core & Aggr Advantage SIA per 100G	96	96	0
FCM-Demo	Core & Aggr Essentials SIA per 100G	96	96	0
FCM-Demo	Core & Aggr Essentials SIA per 100G for Cisco 8000	128	128	16
FCM-Demo	Core & Aggr Essentials SW Right-to-Use per 100G f	126	126	20
FCM-Demo	Core and Aggregation Advantage SW Right-to-Use v	96	96	0
FCM-Demo	Core and Aggregation Essentials SW Right-to-Use v	196	196	0
FCM-Demo	XRd Control Plane	15	15	0

## ライセンス レポートの実行

\* 名前: ライセンス レポート

説明: View License count and usage over selected Virtual

\* バーチャル アカウ  
ント: 選択された仮想アカウント

\* ライセンス: すべてのライセンス

67 選択済み

③ 条件を入力

キーワードの入力

すべて選択 | 選択解除

- API Test Virtual Account
- CS Demo Labs
- DCC-Demo
- dCloud SSM for Campus Network

詳細フィルタの非表示

ソース: ソースの選択

製品ファ:

Customer Name: SKU:

Order Number: Device:

請求先PO#: PAK:

出荷国: 国を選択します。

有効期限: 日付の選択

Alerts: Select Alert

④ CSV or XLS にエクスポート

CSVにエクスポート | XLSにエクスポート | キャンセル

# 製品インスタンスレポート

[レポート] タブにて、管理する製品インスタンスに関するレポートを出力することもできます。

Cisco Software Central > スマートソフトウェアライセンス

## スマートソフトウェアライセンス

アラート | インベントリ | スマートライセンスへの変換 | **レポート** | 設定 | オンプレミス アカウント | アクティビティ

レポート

① レポートを選択

レポート | 使用状況データファイル | ポリシーのレポート生成 | デバイスコントローラの同期ファイル

名前 | 説明

ライセンス | View License count and usage over selected Virtual Account

製品インスタンス | ② 製品インスタンスを選択

**レポート設定が使用されました**

Smart Account Name	/ CS Demo Labs / cs.cisco.com
Report Title	製品インスタンス レポート
Report Description:	バーチャル アカウントおよびタイプ別の製品インスタンス
Generated On:	2024-Mar-03 14:34:33
Report Generated by:	mekawaba@cisco.com

⑤ レポート出力例

**Product Instance Details**

バーチャル アカウント	シリアル番号/ID	lost Number	AC Address	UUID	Device Name	Device PID	Device Description	登録日	最終連絡日	License	製品タイプ	License Quantity Used
loud-mekawaba-202308	FOC2217JHIW	-	-	-	c8k-0	8201	8000 Series	4-Mar-02 13:24	4-Mar-03 13:24	01 Base HW Tracking P	SPFIRE	1
loud-mekawaba-202308	FOC2217EUHN	-	-	-	c8k-1	8201	8000 Series	4-Mar-02 13:24	4-Mar-02 13:24	01 Base HW Tracking P	SPFIRE	1

## 製品タイプ レポート

\* 名前: 製品インスタンス レポート

説明: バーチャル アカウントおよびタイプ別の製品インスタンス

\* バーチャル アカウント: 選択された仮想アカウント | 1 選択済み

製品タイプ: オプションの選択

\* デバイス識別情報: All

\* Registration TimeFrame: All

\* Alerts: All

③ 条件を入力

④ CSV or XLS にエクスポート

CSVにエクスポート | XLSにエクスポート | キャンセル

# 本日の トピック

- ① FCM (Flexible Consumption Model) 概要
- ② SIA/RTM コンプライアンスチェック
- ③ ライセンス・製品インスタンスレポート
- ④ FCM 自動化 1 Smart Licensing APIs
- ⑤ FCM 自動化 2 NETCONF
- ⑥ FCM 自動化 3 EZ-Register
- ⑦ デモ

# Smart Licensing APIs

Smart Accounts & Licensing APIs が提供されており、CSSM / SSM On-prem に対する各種操作を API 経由で実行することができます。Cisco API Developer Portal (<https://apidocs-prod.cisco.com/>) からドキュメントの閲覧が可能です。

The screenshot displays the Cisco API Developer Portal interface. The top navigation bar includes 'Home', 'Explore', 'Console', 'Support', 'Metrics', and 'GraphQL'. The main header reads 'Cisco API Developer Portal' with the tagline 'Your path to creating a digitized experience with Cisco and customers.' Below this is an 'Explore APIs' button. The 'API Domains' section features a grid of icons for various API categories: Commerce APIs, Commerce Renewal APIs, Customer Experience, Smart Accounts & Licensing APIs (highlighted with a red box and an arrow), and Plug & Play Connect APIs. A search bar and 'View' options are also present. On the right, the 'Smart Accounts & Licensing APIs' page is shown, featuring a search bar, a list of API categories (e.g., Commerce APIs, Customer Experience, Smart Accounts & Licensing APIs), and a 'Support' section with contact information for 'Domain Admins'. A list of API categories is also visible on the right side of the page.

# API Access Request

API の使用には、まず下記のように、Access Request を行います。

The screenshot shows the Cisco API Developer Portal interface. The top navigation bar includes 'Home', 'Explore', 'Console', 'Support', 'Metrics', and 'GraphQL'. The main content area is titled 'Smart Accounts' with a version dropdown set to '1.0'. A search bar is located at the top left. The left sidebar contains a navigation menu with categories like 'Smart Accounts & Licensing APIs', 'Software Orders', 'Software NPI', 'API Alerts', 'API Virtual Accounts', 'API Devices', 'API Licenses', and 'API Smart Accounts'. Under 'API Smart Accounts', several API endpoints are listed with their methods: 'Validate User Access V2' (GET), 'Validate User Access V1' (GET), 'Smart Recommendation for Partners' (POST), and 'Smart Accounts Search' (GET). The main content area displays a description of the Smart Accounts APIs and a button labeled 'Access Request'. A modal dialog titled 'Request Access to an API Smart Accounts' is open, showing a 'Your Profile' section with fields for 'Company/Organization Name', 'Full Name', 'Email Address', 'Cisco ID', and 'Phone'. Below this is a 'Reason for API Access: \*' field with the text 'I'd like to access this API for managing our Smart Accounts.' and a 'Submit' button. A support contact 'Support' with email 'kkuchana@cisco.com' is also visible. Three numbered callouts in Japanese are overlaid on the image: ① Access Request をクリック (Click Access Request), ② リクエスト理由を記入 (Enter request reason), and ③ Submit (Submit).

# API Client 作成

次に、API Client を作成し、Client ID と Client Secret を取得します。

**① Console を選択**

**② Create API Client をクリック**

**③ Application 名を入力**

**④ 利用規約に同意**

**⑤ API Services を選択**

**⑥ 関連づける API を選択**

**⑦ Save**

**API Client Registered Successfully**

Client ID:  
40d782fc-fda2-43a4-97da-31134b86783b

Client Secret:  
.....

**Create an API Client**

API Client Application Type Associate APIs

Application Details

Name of your application: \*

FCM demo 3

Application description (optional):

**Note:** The Import Credential option is used to enter OAuth2 credentials for the Okta Auth service. Note the credentials will be validated using client type. To be able to import the credentials, you must have created then type "client\_credential", in addition to other grant type you may have selected.

Import Credentials

I agree to the terms of service

**Create an API Client**

API Client Application Type Associate APIs

Select Application Type: \*

Type	Grant types	Description
<input type="radio"/> Single Page	authorization_code,implicit,refresh_token	Single page application that runs in a browser which maintains a session with the server.
<input type="radio"/> Native	authorization_code,implicit,refresh_token,password	Desktop or Mobile application that redirects user to the server for authentication and maintains a session on the device.
<input type="radio"/> Web	authorization_code,implicit,refresh_token,client_credentials	Service-side application that runs on the server and interacts with the API using OAuth access tokens.
<input checked="" type="radio"/> API Services	client_credentials	Application that interacts with the API using OAuth access tokens.

**Create an API Client**

API Client Application Type Associate APIs

**Note:**

- Associating an API which is a member of a package will associate all APIs in that package. Likewise, removing an API which is a member of a package will unassociate all APIs in the package.
- Only APIs which share the same authentication service can be associated to an application.
- When associating an API, all APIs which do not use the same auth service will be removed from the list.

Available APIs

Search by Package or API Name

Package	API Name
No APIs available	

Assign Selected

Unassign Selected

Unassign All

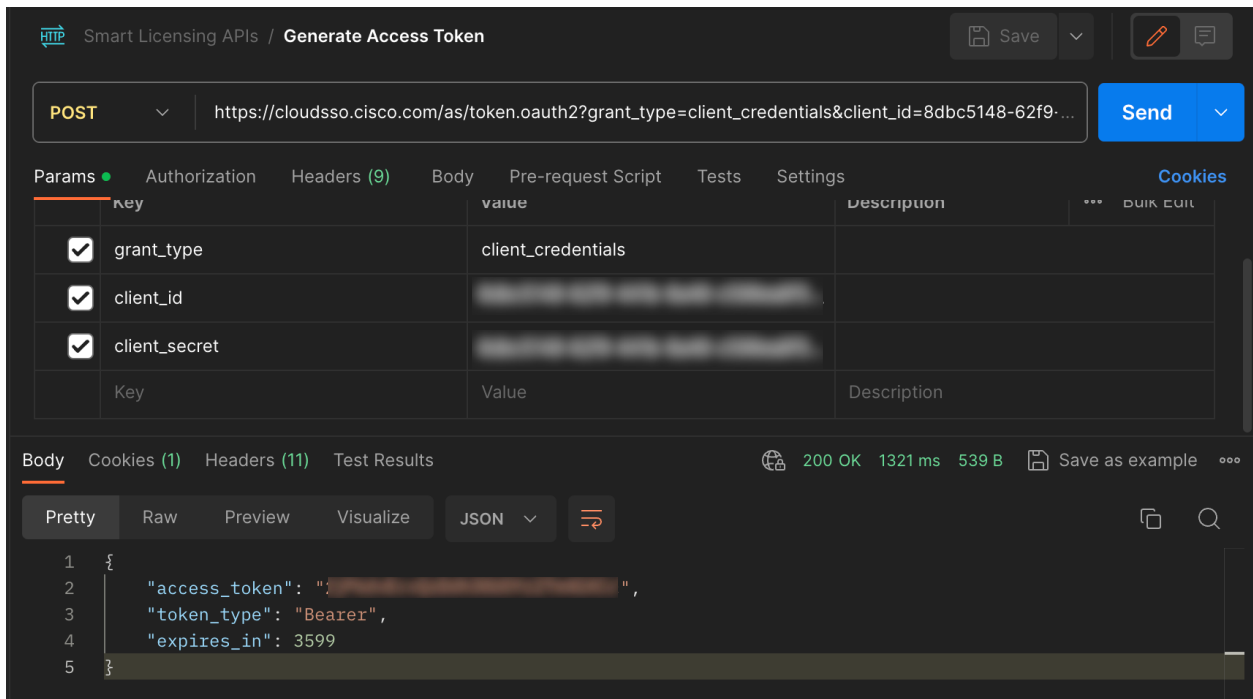
APIs to Associate

Search by API Name

- Software
- Alerts
- Virtual Accounts
- Devices
- Licenses
- Secret Accounts
- Tokens

# Bearer Token の生成

Client ID と Client Secret を用いて、Bearer Token を生成します。下記に Postman を利用し、Token API を呼び出す例を記載します。アクセストークンは応答で指定された期間が経過すると期限切れになることにご注意ください。



The screenshot shows a Postman interface for a POST request to generate an access token. The URL is `https://cloudsso.cisco.com/as/token.oauth2?grant_type=client_credentials&client_id=8dbc5148-62f9-...`. The request parameters are:

key	value	Description
<input checked="" type="checkbox"/> grant_type	client_credentials	
<input checked="" type="checkbox"/> client_id	[REDACTED]	
<input checked="" type="checkbox"/> client_secret	[REDACTED]	

The response is a JSON object:

```
1 {
2   "access_token": "[REDACTED]",
3   "token_type": "Bearer",
4   "expires_in": 3599
5 }
```

- Method: POST
- Request URL: <https://cloudsso.cisco.com/as/token.oauth2>
- Params:
  - grant\_type (client\_credentials)
  - client\_id
  - client\_secret

# Smart Licensing APIs 使用例

生成したアクセストークンを用いて、ライセンス使用状況、アラート情報、製品インスタンスリストなどを取得できます。また、バーチャルアカウントの作成、製品登録トークンの生成なども可能です。詳細は、Cisco API Developer Portal をご確認ください。

POST `https://swapi.cisco.com/services/api/smart-accounts-and-licensing/v1/accounts/{smart_account_id}/licenses`

Body (JSON):

```

1 {
2   "virtualAccounts": ["FCM-Demo"]
3 }

```

Smart Account / Virtual Account (option) が保有するライセンス及び使用状況の取得

Body (Pretty):

```

1 {
2   "totalRecords": 10,
3   "licenses": [
4     {
5       "isPortable": false,
6       "licenseSubstitutions": [],
7       "quantity": 7,
8       "ahaApps": false,
9       "available": 2,
10      "license": "8201 Base HW Tracking PID",
11      "licenseDetails": [
12        {
13          "licenseType": "TERM",
14          "quantity": 7,
15          "licenseSku": "8201-TRK",
16          "endDate": "2033-06-03",
17          "subscriptionId": null
18        }
19      ]
20     }
21   ]
22 }

```

POST `https://swapi.cisco.com/services/api/smart-accounts-and-licensing/v1/accounts/{smart_account_id}/alerts`

Body (JSON):

```

1 {
2   "virtualAccounts": ["FCM-Demo"],
3   "severity": ["Minor"]
4 }

```

Smart Account / Virtual Account (option) において発生中のアラート情報の取得

Body (Pretty):

```

1 {
2   "status": "SUCCESS",
3   "statusMessage": "",
4   "totalRecords": 34,
5   "alerts": [
6     {
7       "licenseEndDate": "Mar 05, 2023",
8       "licenseStartDate": "Mar 6, 2022",
9       "virtualAccount": "FCM-Demo",
10      "message": "2 \\Core & Aggr Advantage SIA per 100G\\ term license Virtual Account \\FCM-Demo\\ expired on Mar 05, 2023",
11      "license": "Core & Aggr Advantage SIA per 100G",
12      "severity": "Minor",
13      "messageType": "Licenses Expired",
14      "actionDue": "Now",
15      "source": "FCM-Demo",
16      "sourceType": "Virtual Account"
17     }
18   ]
19 }

```

POST `https://swapi.cisco.com/services/api/smart-accounts-and-licensing/v1/accounts/{smart_account_id}/devices`

Body (JSON):

```

1 {
2   "virtualAccounts": ["FCM-Demo"]
3 }

```

Smart Account / Virtual Account (option) に登録された製品インスタンス情報の取得

Body (Pretty):

```

3 {
4   "devices": [
5     {
6       "hostName": "BRU-N55A2-SE-2",
7       "displayName": "BRU-N55A2-SE-2",
8       "productName": "Network Convergence System 5500",
9       "mode": "SL",
10      "licenseUsage": [],
11      "sudi": {
12        "macAddress": null,
13        "suvi": null,
14        "hostIdentifier": null,
15        "udiSerialNumber": "F0C2304R20T",
16        "udiPid": "NCS-55A2-MOD-SE-S",
17        "uuid": null,
18        "udiVid": null
19      }
20     }
21   ]
22 }

```



# 本日の トピック

- ① FCM (Flexible Consumption Model) 概要
- ② SIA/RTM コンプライアンスチェック
- ③ ライセンス・製品インスタンスレポート
- ④ FCM 自動化 1 Smart Licensing APIs
- ⑤ FCM 自動化 2 NETCONF
- ⑥ FCM 自動化 3 EZ-Register
- ⑦ デモ

# Smart Licence 関連の YANG モデル

下記の YANG モデルを用いて、Smart License 関連の操作を XR デバイスに対して実行できます。

タイプ	YANG モデル	対応する CLI コマンド
Configuration Data	Native data model: <ul style="list-style-type: none"> <li>Cisco-IOS-XR-smart-licensecfg.yang</li> </ul>	<ul style="list-style-type: none"> <li>license smart reservation</li> <li>[no] license smart reservation</li> <li>license smart flexibleconsumption enable</li> <li>[no] license smart flexibleconsumption enable</li> </ul>
Operations Data	Common data model: <ul style="list-style-type: none"> <li>cisco-smart-license.yang</li> </ul> Native data model: <ul style="list-style-type: none"> <li>Cisco-IOS-XR-smart-licenseplatform-oper.yang</li> <li>Cisco-IOS-XR-infra-smartlicenseoper.yang</li> </ul>	<ul style="list-style-type: none"> <li>show license platform summary</li> <li>show license platform detail</li> <li>show license [all   summary   usage   udi]</li> </ul>
Actions	Common data model: <ul style="list-style-type: none"> <li>cisco-smart-license.yang</li> </ul> Native data model: <ul style="list-style-type: none"> <li>Cisco-IOS-XR-smart-licenseplatform-oper.yang</li> <li>Cisco-IOS-XR-infra-smartlicenseoper.yang</li> </ul>	<ul style="list-style-type: none"> <li>license smart register id token</li> <li>license smart deregister</li> <li>license smart renew id</li> <li>license smart renew auth</li> <li>license smart reservation request local</li> <li>license smart reservation cancel local</li> <li>license smart reservation install file</li> <li>license smart reservation return local</li> <li>license smart reservation return authorization file</li> <li>license smart transport smart</li> <li>license smart url</li> <li>license smart software-upgrade enable</li> <li>license smart proxy hostname</li> <li>license smart proxy port</li> </ul>

# netconf-console2 による NETCONF テスト

netconf-console2 は、Python ベースの NETCONF テストツールです。

netconf-console2 のインストール

```
$ pip install netconf-console2
```

NETCONF によるデバイス登録の実施例

sl\_register\_token.xml

```
<register-id-token xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-smart-license-act">  
  <id-token>SSM YOUR TOKEN HERE</id-token>  
</register-id-token>
```

NETCONF RPC Call

```
$ netconf-console2 --host=198.18.134.60 --port 830 -u cisco -p ***** --rpc=sl_register_token.xml  
  
<?xml version='1.0' encoding='UTF-8'?>  
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"  
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:9a2557f7-ce25-452a-a392-  
87e2149af8c6">  
  <response xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-smart-license-act">  
    <operation-status>success</operation-status>  
    <return-code>registrationinprogress</return-code>  
  </response>  
</rpc-reply>
```

# Python による自動化例: FCM ステータスの確認

下記は、Python ncclient library を用いて XR デバイスの FCM ステータスを確認するスクリプト例です。

```
#!/usr/bin/env python
# sl_report.py

from ncclient import manager
import xmltodict

NETCONF_PORT = "830"
USERNAME = "cisco"
PASSWORD = "*****"
HOSTS = ["198.18.134.60", "198.18.134.61"]

get_filter_auth = '''
<smart-agent xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-infra-smartlicense-oper">
  <licensing>
    <state>
      <state-info>
        <authorization>
          <authorization-state/>
        </authorization>
      </state-info>
    </state>
  </licensing>
</smart-agent>
'''

get_filter_registration = '''
<smart-agent xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-infra-smartlicense-oper">
  <licensing>
    <state>
      <state-info>
        <registration>
          <registration-state/>
        </registration>
      </state-info>
    </state>
  </licensing>
</smart-agent>
'''
```

sl\_report.py

```
def fcm_report(HOST):
    with manager.connect(
        host=HOST,
        port=NETCONF_PORT,
        username=USERNAME,
        password=PASSWORD,
        hostkey_verify=False, device_params={"name":"iosxr"}) as device:

        reply = device.get(filter=("subtree", get_filter_registration)).data_xml
        xml_dict = xmltodict.parse(reply)
        register_state = xml_dict["data"]["smart-agent"]["licensing"]["state"]["state-info"]["registration"]["registration-state"]

        reply = device.get(filter=("subtree", get_filter_auth)).data_xml
        xml_dict = xmltodict.parse(reply)
        auth_state = xml_dict["data"]["smart-agent"]["licensing"]["state"]["state-info"]["authorization"]["authorization-state"]

        print(f"{HOST:<15} | {register_state :<25} | {auth_state:<40}")

if __name__ == '__main__':
    print(f"{'Device':<15} | {'FCM Registration':<25} | {'FCM Status':<40}")
    print("-----")
    for h in HOSTS:
        fcm_report(h)
```

sl\_report.py (続)

```
cisco@ubuntu:~/fcm/FCM/ncpy$ python3 sl_report_test.py
```

実行例

Device	FCM Registration	FCM Status
198.18.134.60	complete	authorized
198.18.134.61	not-registered	eval

# 本日の トピック

- ① FCM (Flexible Consumption Model) 概要
- ② SIA/RTM コンプライアンスチェック
- ③ ライセンス・製品インスタンスレポート
- ④ FCM 自動化 1 Smart Licensing APIs
- ⑤ FCM 自動化 2 NETCONF
- ⑥ FCM 自動化 3 EZ-Register
- ⑦ デモ

# EZ-Register とは

EZ-Register とは、CSSM / SSM On-Premise でのデバイス登録トークンの生成からデバイス登録までを自動化するツールです。Github で公開されており、多数のデバイスの一括登録を簡単に実行できます。

## EZ-Register 利用の流れ



「④ FCM 自動化 1 Smart Licensing APIs」にて説明

### EZ-Register のクローン

```
$ git clone https://github.com/ios-xr/ez-register-gui
```

### 必要パッケージのインストール

```
$ cd ez-register-gui  
$ pip install -r requirements.txt
```

# EZ-Register Input File の作成

EZ-Register にて使用する Input File はテンプレート (ez-register-gui/input\_file\_templates) を使用し、下記の項目を入力します。

ez\_register\_direct.xls (CSSM へのダイレクト接続) の記入例

Hostname	Username	Password	Smart Account	Virtual Account	FCM	Token Description
198.18.134.60	cisco	*****	cs.cisco.com	FCM-Demo	Y	EZR Script
Token Expiration (Days)	Token Export Controlled	Client ID	Client Secret	VRF	Re-Register	Device Name
30	Allowed	689cd297-1...	8dea68cf-1...	Default	N	c8k-1

# EZ-Register の実行

Input File を指定し、EZ-Register を下記のように実行します。

## EZ-Register の実行例

```
$ ./ez_register_direct.py ./ez_register_direct.xls
Beginning Registration Attempts
register vrf :
Host: 198.18.134.60 - Registration attempt completed

Beginning Verification
Host: 198.18.134.60 - Registration Successful

Out of 1 node(s), 1 node(s) is/are successfully registered
```

## EZ-Register のログ出力例

```
$ less ez_register_direct_20240302_152657.log
INFO 2024-03-02 15:27:16,647 - =====
INFO 2024-03-02 15:27:16,647 - Creating access token to securely connect CSSM
INFO 2024-03-02 15:27:16,647 - =====
INFO 2024-03-02 15:27:18,933 - {"error_description":"Invalid client or client
credentials.,"error":"invalid_client"}
INFO 2024-03-02 15:27:18,934 - =====
INFO 2024-03-02 15:27:18,934 - Exception!!
INFO 2024-03-02 15:27:18,934 - =====
INFO 2024-03-02 15:27:18,934 - 'access_token'
```



# 本日の トピック

- ① FCM (Flexible Consumption Model) 概要
- ② SIA/RTM コンプライアンスチェック
- ③ ライセンス・製品インスタンスレポート
- ④ FCM 自動化 1 Smart Licensing APIs
- ⑤ FCM 自動化 2 NETCONF
- ⑥ FCM 自動化 3 EZ-Register
- ⑦ デモ

# 本日のポイント



1

Flexible Consumption Model (FCM) は、ハードウェアのキャパシティ(帯域幅)やソフトウェアの機能をニーズに合わせて導入、投資を最適化できる購入モデルです。

2

CSSM にて、保有するライセンスのインベントリ、ライセンスの使用状況 (過不足)、およびアラートを確認できます。

3

CSSM にて、保有するライセンス及び管理する製品インスタンスに関するレポートを出力できます。

4

Smart Accounts APIs が提供されており、CSSM / SSM On-prem に対する各種操作を API 経由で実行することができます。

5

NETCONF 経由で Smart License 関連の XR デバイスに対する操作を自動化できます。

6

EZ-Register で、CSSM / SSM On-Premise でのデバイス登録トークンの生成からデバイス登録までを自動化可能です。

# 参考リンク

- Cisco Software Central  
<https://software.cisco.com/>
- Cisco API Developer Portal  
<https://apidocs-prod.cisco.com/>
- Cisco Smart Licensing Guide  
<https://www.cisco.com/c/en/us/buy/licensing/licensing-guide.html>



# Q&A



